Computer Graphics (CS 4731)
Lecture 11: Linear Algebra for Graphics (Points, Scalars, Vectors)

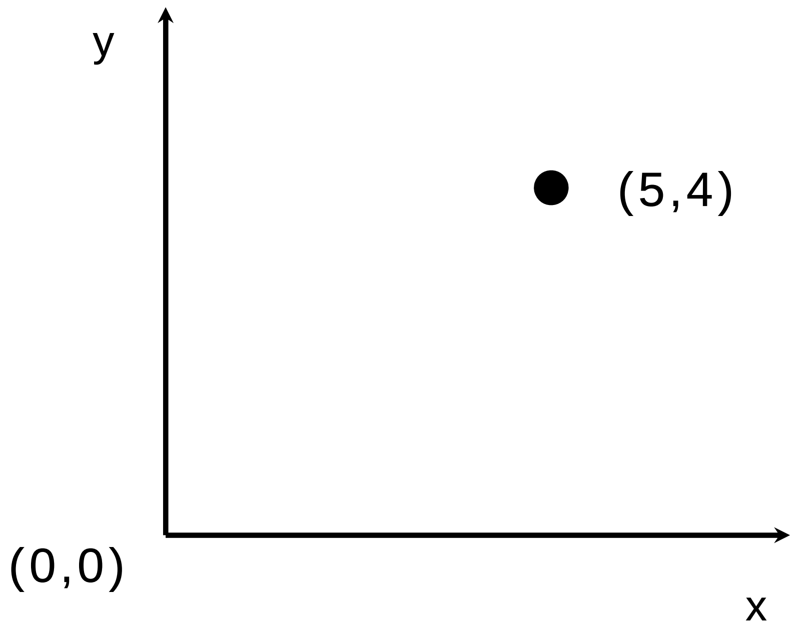
Prof Emmanuel Agu

Computer Science Dept.

Worcester Polytechnic Institute (WPI)

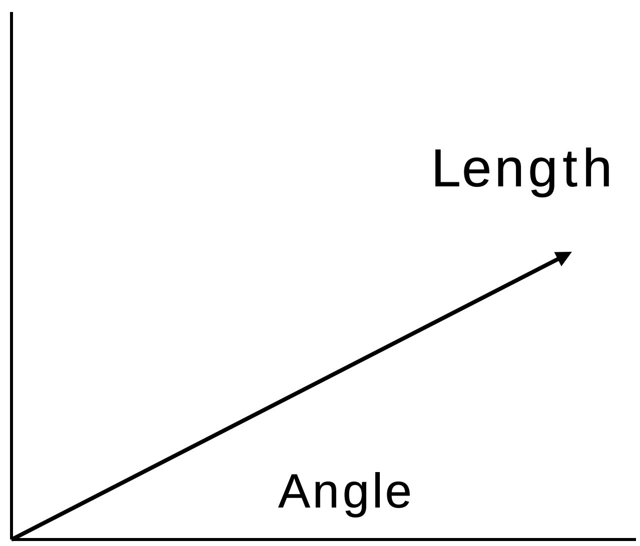
Points, Scalars and Vectors

- Points, vectors defined relative to a coordinate system
- Example: Point (5,4)



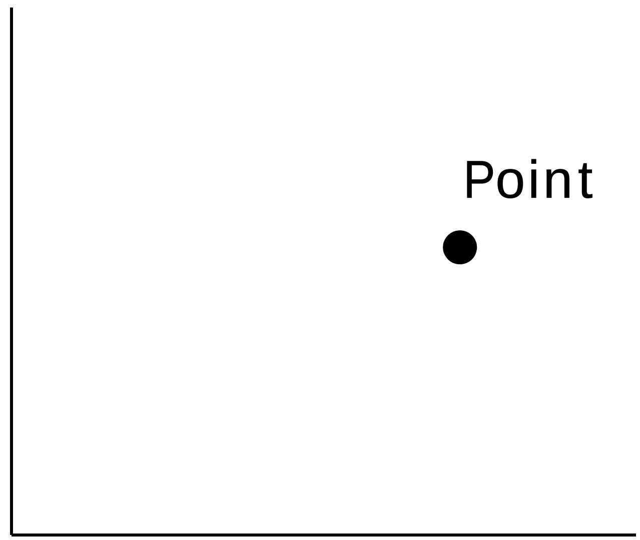
Vectors

- Magnitude
- Direction
- **NO** position
- Can be added, scaled, rotated
- CG vectors: 2, 3 or 4 dimensions



Points

- Location in coordinate system
- Cannot add or scale
- Subtract 2 points = vector



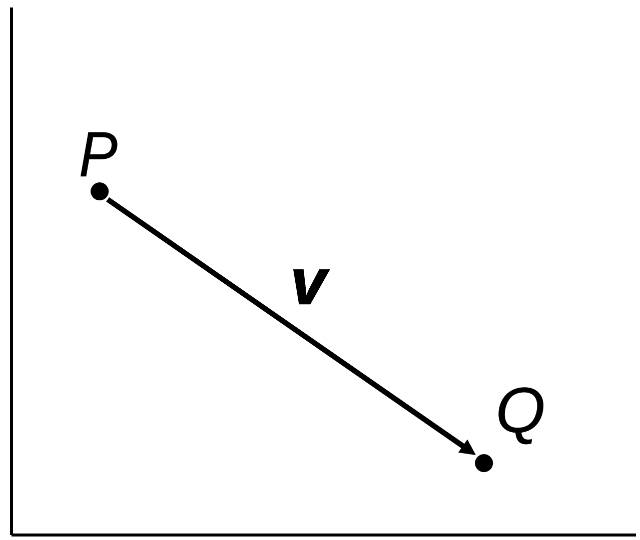
Vector-Point Relationship

- Diff. b/w 2 points = vector

$$\mathbf{v} = Q - P$$

- point + vector = point

$$\mathbf{v} + P = Q$$



Vector Operations

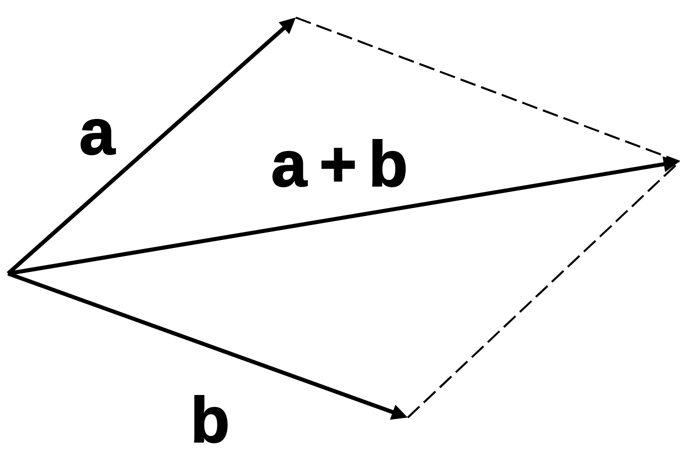
- Define vectors

$$\mathbf{a} = (a_{1,} a_2, a_3)$$

$$\mathbf{b} = (b_{1,} b_2, b_3)$$

Then vector addition:

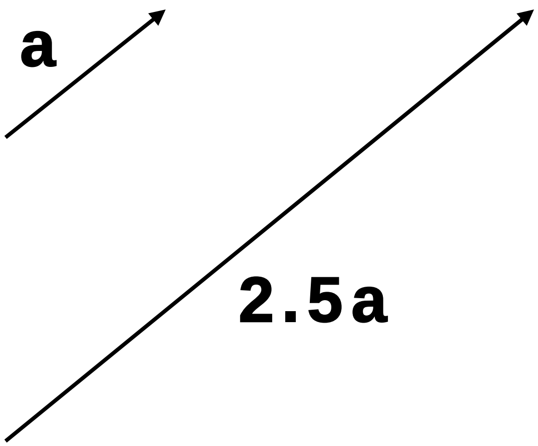
$$\mathbf{a} + \mathbf{b} = (a_1 + b_{1,} a_2 + b_2, a_3 + b_3)$$



Vector Operations

- Define scalar, *s*
- Scaling vector by a scalar

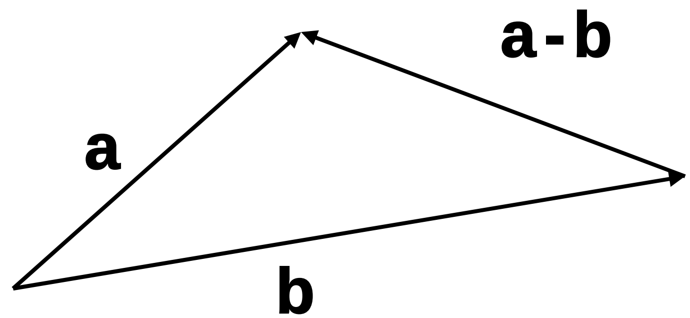
$$\mathbf{a}s = (a_1 s, a_2 s, a_3 s)$$



Note vector subtraction:

$$\mathbf{a} - \mathbf{b}$$

$$= (a_1 + (-b_1), a_2 + (-b_2), a_3 + (-b_3))$$



Vector Operations: Examples

- Scaling vector by a scalar

$$\mathbf{a}s = (a_1 s, a_2 s, a_3 s)$$

- Vector addition:

$$\mathbf{a} + \mathbf{b} = (a_1 + b_{1,} a_2 + b_2, a_3 + b_3)$$

- For example, if **a**=(2,5,6) and **b**=(-2,7,1) and *s*=6, then

$$\mathbf{a} + \mathbf{b} = (a_1 + b_{1,} a_2 + b_2, a_3 + b_3) = (0,12,7)$$

$$\mathbf{a}s = (a_1 s, a_2 s, a_3 s) = (12,30,36)$$

Affine Combination

- Given a vector

$$\mathbf{a} = (a_{1,} a_2, a_3, \ldots, a_n)$$

$$a_1 + a_2 + \ldots\ldots\ldots a_n = 1$$

- Affine combination: Sum of all components = 1

- Convex affine = affine + no negative component i.e

$$a_1, a_2, \ldots\ldots\ldots a_n = non - negative$$

Magnitude of a Vector

- Magnitude of **a**

$$|\mathbf{a}| = \sqrt{a_1{}^2 + a_2{}^2 .... + a_n{}^2}$$

- Normalizing a vector (unit vector)

$$\hat{\mathbf{a}} = \frac{\mathbf{a}}{|\mathbf{a}|} = \frac{vector}{magnitude}$$

- Note magnitude of normalized vector = 1. i.e

$$\sqrt{a_1{}^2 + a_2{}^2 .... + a_n{}^2} = 1$$

Magnitude of a Vector

- Example: if a = (2, 5, 6)

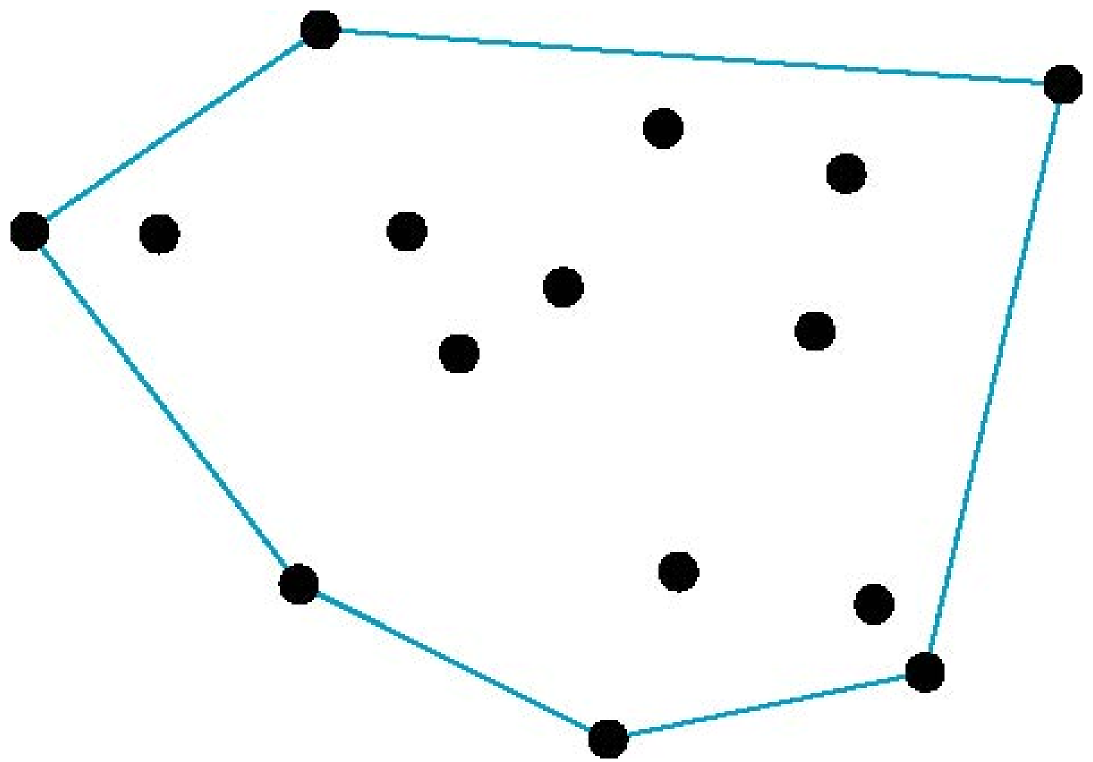
- Magnitude of **a** $$|\mathbf{a}| = \sqrt{2^2 + 5^2 + 6^2} = \sqrt{65}$$

- Normalizing **a**

$$\hat{\mathbf{a}} = \left(\frac{2}{\sqrt{65}}, \frac{5}{\sqrt{65}}, \frac{6}{\sqrt{65}} \right)$$

Convex Hull

- Smallest convex object containing $P_1, P_2, \ldots\ldots P_n$
- Formed by “shrink wrapping” points



Dot Product (Scalar product)

- Dot product,

$$d = \mathbf{a} \cdot \mathbf{b} = a_1 \cdot b_1 + a_2 \cdot b_2 .... + a_3 \cdot b_3$$

- For example, if *a*=(2,3,1) and *b*=(0,4,-1) then

$$a \cdot b = (2 \times 0) + (3 \times 4) + (1 \times -1)$$

$$= 0 + 12 - 1 = 11$$

Properties of Dot Products

- Symmetry (or commutative):

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a}$$

- Linearity:

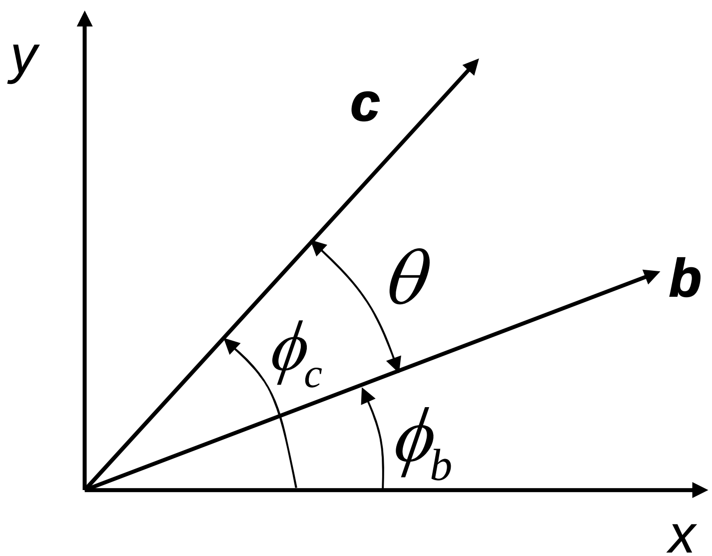
$$(\mathbf{a} + \mathbf{c}) \cdot \mathbf{b} = \mathbf{a} \cdot \mathbf{b} + \mathbf{c} \cdot \mathbf{b}$$

- Homogeneity:

$$(s\mathbf{a}) \cdot \mathbf{b} = s(\mathbf{a} \cdot \mathbf{b})$$

- And $\left|\mathbf{b}^2\right| = \mathbf{b} \cdot \mathbf{b}$

Angle Between Two Vectors

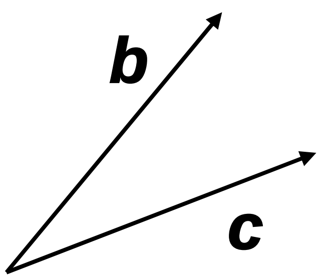


$$\mathbf{b} = \left(|\mathbf{b}|\cos\phi_b, |\mathbf{b}|\sin\phi_b\right)$$

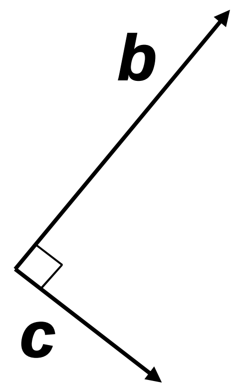
$$\mathbf{c} = \left(|\mathbf{c}|\cos\phi_c, |\mathbf{c}|\sin\phi_c\right)$$

$$\mathbf{b}\cdot\mathbf{c} = |\mathbf{b}||\mathbf{c}|\cos\theta$$

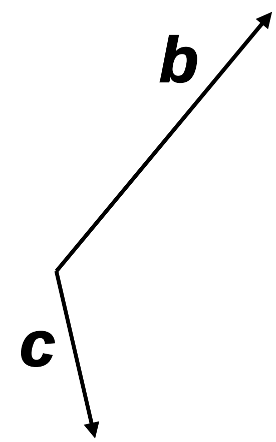
Sign of **b.c**:



b.c > 0



b.c = 0



b.c < 0

Angle Between Two Vectors

- Find the angle b/w the vectors **b** = (3,4) and **c =** (5,2)

Angle Between Two Vectors

- Find the angle b/w the vectors **b** = (3,4) and **c = (5,2)**
  - |**b**|= 5, |**c**|= 5.385

$$\hat{\mathbf{b}} = \left(\frac{3}{5}, \frac{4}{5}\right) \qquad \hat{\mathbf{b}} \bullet \hat{\mathbf{c}} = 0.85422 = \cos\theta$$

$$\theta = 31.326^\circ$$

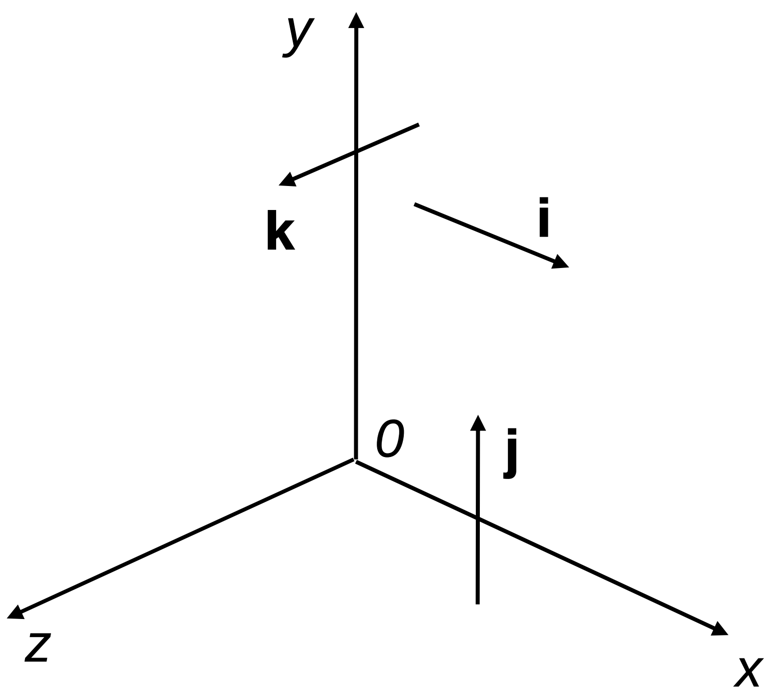
Standard Unit Vectors

Define

$$\mathbf{i} = (1,0,0)$$

$$\mathbf{j} = (0,1,0)$$

$$\mathbf{k} = (0,0,1)$$



So that any vector,

$$\mathbf{v} = (a,b,c) = a\mathbf{i} + b\mathbf{j} + c\mathbf{k}$$

Cross Product (Vector product)

If

$$\mathbf{a} = (a_x, a_y, a_z) \qquad \mathbf{b} = (b_x, b_y, b_z)$$

Then

$$\mathbf{a} \times \mathbf{b} = (a_y b_z - a_z b_y)\mathbf{i} - (a_x b_z - a_z b_x)\mathbf{j} + (a_x b_y - a_y b_x)\mathbf{k}$$

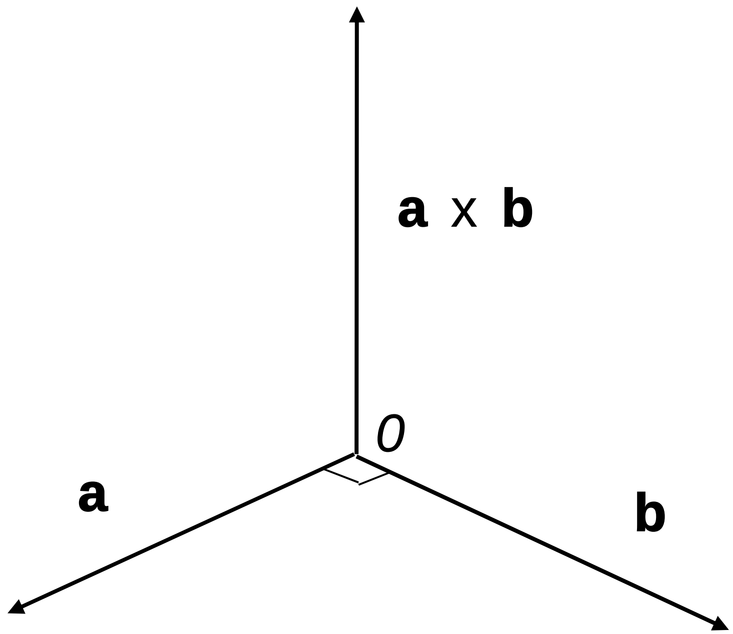
Remember using determinant

$$\begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix}$$

Note: a x **b** is perpendicular to **a** and **b**

Cross Product

Note: a x **b** is perpendicular to both **a** and **b**



Cross Product

Calculate **a x b** if a = (3,0,2) and **b** = (4,1,8)

Cross Product

Calculate **a x b** if a = (3,0,2) and **b** = (4,1,8)

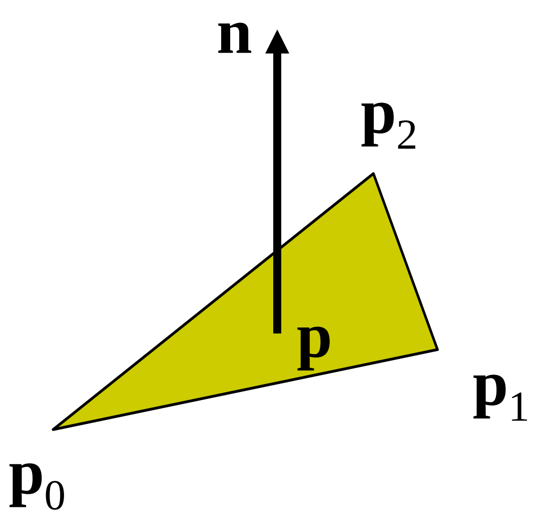
a x b = -2**i** – 16**j** + 3**k**

Normal for Triangle using Cross Product Method

plane $\mathbf{n} \cdot (\mathbf{p} - \mathbf{p}_0) = 0$

$\mathbf{n} = (\mathbf{p}_2 - \mathbf{p}_0) \times (\mathbf{p}_1 - \mathbf{p}_0)$

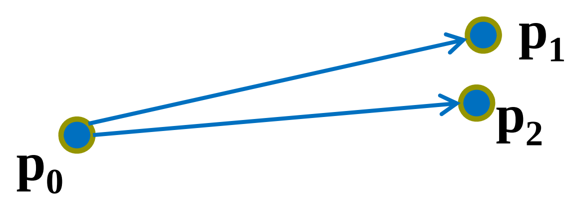
normalize $\mathbf{n} \leftarrow \mathbf{n} / |\mathbf{n}|$



Note that right-hand rule determines outward face

Newell Method for Normal Vectors

- Problems with cross product method:
  - calculation difficult by hand, tedious
  - If 2 vectors almost parallel, cross product is small
  - Numerical inaccuracy may result

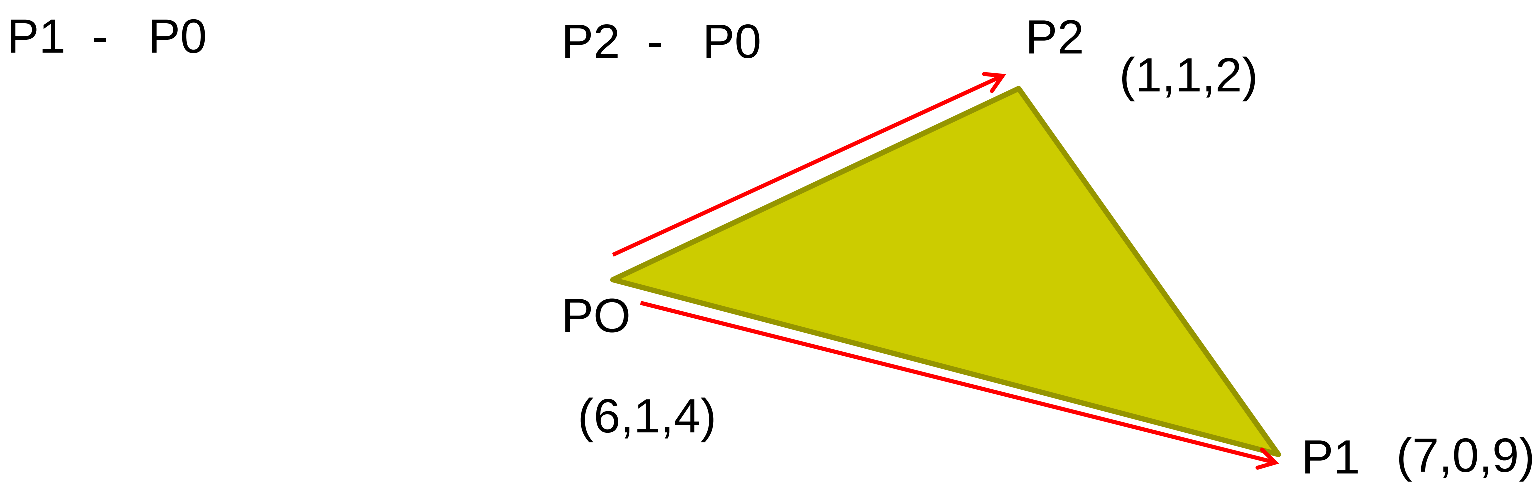


- Proposed by Martin Newell at Utah (teapot guy)
  - Uses formulae, suitable for computer
  - Compute during mesh generation
  - Robust!

Newell Method Example

- Example: Find normal of polygon with vertices P0 = (6,1,4), P1=(7,0,9) and P2 = (1,1,2)

- Using simple cross product: ((7,0,9)-(6,1,4)) X ((1,1,2)-(6,1,4)) = (2,-23,-5)



Newell Method for Normal Vectors

- Formulae: Normal N = (mx, my, mz)

$$m_x = \sum_{i=0}^{N-1} \left(y_i - y_{next(i)}\right)\left(z_i + z_{next(i)}\right)$$

$$m_y = \sum_{i=0}^{N-1} \left(z_i - z_{next(i)}\right)\left(x_i + x_{next(i)}\right)$$

$$m_z = \sum_{i=0}^{N-1} \left(x_i - x_{next(i)}\right)\left(y_i + y_{next(i)}\right)$$

Using Newell method, for previous example plug in values result is same:
Normal is (2, -23, -5)

References

- Angel and Shreiner, Interactive Computer Graphics, 6th edition, Chapter 3
- Hill and Kelley, Computer Graphics using OpenGL, 3rd edition, Sections 4.2 - 4.4